CSV-backed To-Do List Python Program (todo_csv.py)

```
# todo_csv.py
import csv
import os
from datetime import datetime

FILENAME = "todo.csv"
FIELDNAMES = ["Task", "CreatedAt", "Completed"]

def load_tasks():
    """Return a list of dicts representing tasks."""
    if not os.path.exists(FILENAME):
        return []
    with open(FILENAME, mode="r", newline="", encoding="utf-8") as f:
        reader = csv.DictReader(f, fieldnames=FIELDNAMES)
        return [row for row in reader]

def save_tasks(tasks):
    """Write list of dicts to CSV (overwrites file)."""
    with open(FILENAME, mode="w", newline="", encoding="utf-8") as f:
        writer = csv.DictWriter(f, fieldnames=FIELDNAMES)
        for task in tasks:
            writer.writerow(task)

def show_menu():
    print("\n----- TO-DO LIST (CSV) -----")
    print("1. Add Task")
    print("2. View Tasks")
    print("3. Remove Task")
    print("4. Mark Task Completed")
    print("5. Exit")

def add_task(tasks):
    text = input("Enter a new task: ").strip()
    if not text:
        print("□ Task cannot be empty.")
        return
    task = {
        "Task": text,
        "CreatedAt": datetime.now().strftime("%Y-%m-%d %H:%M:%S"),
        "Completed": "No"
    }
    tasks.append(task)
    save_tasks(tasks)
    print(f"□ Task '{text}' added.")

def view_tasks(tasks):
    if not tasks:
        print("□ No tasks found.")
        return
    print("\nYour To-Do List:")
    for i, t in enumerate(tasks, start=1):
        status = "✓" if t.get("Completed","No").lower() == "yes" else " "
        print(f"{i}. [{status}] {t.get('Task')} (Added: {t.get('CreatedAt')})")

def remove_task(tasks):
```

```
    if not tasks:
        print("□ No tasks to remove.")
        return
    view_tasks(tasks)
    try:
        n = int(input("Enter task number to remove: "))
        if n < 1 or n > len(tasks):
            raise IndexError
        removed = tasks.pop(n-1)
        save_tasks(tasks)
        print(f"□□ Removed: {removed.get('Task')}")
    except ValueError:
        print("□ Please enter a number.")
    except IndexError:
        print("□ Invalid task number.")

def mark_completed(tasks):
    if not tasks:
        print("□ No tasks to mark.")
        return
    view_tasks(tasks)
    try:
        n = int(input("Enter task number to mark completed: "))
        if n < 1 or n > len(tasks):
            raise IndexError
        tasks[n-1]["Completed"] = "Yes"
        save_tasks(tasks)
        print(f"□ Marked completed: {tasks[n-1].get('Task')}")
    except ValueError:
        print("□ Please enter a number.")
    except IndexError:
        print("□ Invalid task number.")

def main():
    tasks = load_tasks()
    while True:
        show_menu()
        choice = input("Enter choice (1-5): ").strip()
        if choice == "1":
            add_task(tasks)
        elif choice == "2":
            view_tasks(tasks)
        elif choice == "3":
            remove_task(tasks)
        elif choice == "4":
            mark_completed(tasks)
        elif choice == "5":
            print("□ Exiting. All tasks saved to todo.csv")
            break
        else:
            print("□ Invalid choice. Try again.")

if __name__ == "__main__":
    main()
```